

PATENT
8009-1014

IN THE U.S. PATENT AND TRADEMARK OFFICE

In re application of: Keisuke KANAMARU et al.

Conf.:

Appl. No.:

Group:

Filed: July 15, 2003

Examiner:

Title: DELAY OPTIMIZATION DESIGNING SYSTEM AND
DELAY OPTIMIZATION DESIGNING METHOD FOR
A LOGIC CIRCUIT AND CONTROL PROGRAM

CLAIM TO PRIORITY

Assistant Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

July 15, 2003

Sir:

Applicant(s) herewith claim(s) the benefit of the
priority filing date of the following application(s) for the
above-entitled U.S. application under the provisions of 35
U.S.C. § 119 and 37 C.F.R. § 1.55:

| <u>Country</u> | <u>Application No.</u> | <u>Filed</u> |
|----------------|------------------------|---------------|
| JAPAN | 2002-207658 | July 17, 2002 |

Certified copy(ies) of the above-noted application(s)
is(are) attached hereto.

Respectfully submitted,

YOUNG & THOMPSON



Benoit Castel, Reg. No. 35,041

745 South 23rd Street
Arlington, VA 22202
Telephone (703) 521-2297

BC/baf

Attachment(s): 1 Certified Copy(ies)

日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2002年 7月17日

出 願 番 号

Application Number:

特願2002-207658

[ST.10/C]:

[JP2002-207658]

出 願 人

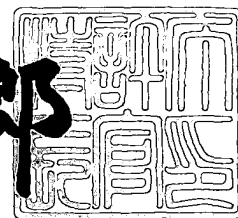
Applicant(s):

日本電気株式会社

2003年 5月 6日

特 許 庁 長 官
Commissioner,
Japan Patent Office

太田 信一郎



出証番号 出証特2003-3032922

【書類名】 特許願

【整理番号】 66206662

【提出日】 平成14年 7月17日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 15/00

【発明者】

 【住所又は居所】 東京都港区芝五丁目7番1号 日本電気株式会社内

 【氏名】 金丸 恵祐

【発明者】

 【住所又は居所】 東京都港区芝五丁目7番1号 日本電気株式会社内

 【氏名】 吉川 浩

【特許出願人】

 【識別番号】 000004237

 【氏名又は名称】 日本電気株式会社

【代理人】

 【識別番号】 100088812

 【弁理士】

 【氏名又は名称】 ▲柳▼川 信

【手数料の表示】

 【予納台帳番号】 030982

 【納付金額】 21,000円

【提出物件の目録】

 【物件名】 明細書 1

 【物件名】 図面 1

 【物件名】 要約書 1

 【包括委任状番号】 9001833

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 論理回路の遅延最適化システムおよびその最適化方法ならびに制御プログラム

【特許請求の範囲】

【請求項 1】 フリップフロップと、組み合わせ回路とを含む論理回路の遅延最適化システムであって、

ラッチに変換しない前記フリップフロップを選定するラッチ化フリップフロップ選定手段と、

前記ラッチ化フリップフロップ選定手段にて選定されなかった前記フリップフロップを、入力側の遅延余裕を出力側に貸し出すことが可能なスルー状態を有するラッチに変換するラッチ変換手段とを含むことを特徴とする論理回路の遅延最適化システム。

【請求項 2】

前記フリップフロップはクロックにおける第 1 レベルから第 2 レベルへの変化時のエッジに応じて入力されるデータを保持するものであり、前記ラッチ変換手段におけるラッチは前記クロックが前記第 1 のレベルを示している際の入力データを出力側にスルーするものであることを特徴とする請求項 1 記載の論理回路の遅延最適化システム。

【請求項 3】

前記ラッチ変換手段におけるラッチとは異なる時間にスルー状態となる第 2 のラッチを論理回路の所定部分に挿入するラッチ挿入手段を含むことを特徴とする請求項 1 または 2 記載の論理回路の遅延最適化システム。

【請求項 4】

前記ラッチ挿入手段における第 2 のラッチは前記クロックが前記第 2 のレベルを示している際の入力データを出力側にスルーするものであることを特徴とする請求項 3 記載の論理回路の遅延最適化システム。

【請求項 5】

前記ラッチ挿入手段における第 2 のラッチは、前記ラッチ変換手段における

ラッチの出力側に設けられることを特徴とする請求項 3 または 4 記載の論理回路の遅延最適化システム。

【請求項 6】 フリップフロップと、組み合わせ回路とを含む論理回路の遅延最適化方法であって、

ラッチに変換しない前記フリップフロップを選定するラッチ化フリップフロップ選定ステップと、

前記ラッチ化フリップフロップ選定ステップにて選定されなかった前記フリップフロップを、入力側の遅延余裕を出力側に貸し出すことが可能なスルー状態を有するラッチに変換するラッチ変換ステップとを含むことを特徴とする論理回路の遅延最適化方法。

【請求項 7】

前記フリップフロップはクロックにおける第 1 レベルから第 2 レベルへの変化時のエッジに応じて入力されるデータを保持するものであり、前記ラッチ変換ステップにおけるラッチは前記クロックが前記第 1 のレベルを示している際の入力データを出力側にスルーするものであることを特徴とする請求項 6 記載の論理回路の遅延最適化方法。

【請求項 8】

前記ラッチ変換ステップにおけるラッチとは異なる時間にスルー状態となる第 2 のラッチを論理回路の所定部分に挿入するラッチ挿入ステップを含むことを特徴とする請求項 6 または 7 記載の論理回路の遅延最適化方法。

【請求項 9】

前記ラッチ挿入ステップにおける第 2 のラッチは前記クロックが前記第 2 のレベルを示している際の入力データを出力側にスルーするものであることを特徴とする請求項 8 記載の論理回路の遅延最適化方法。

【請求項 10】

前記ラッチ挿入ステップにおける第 2 のラッチは、前記ラッチ変換ステップにおけるラッチの出力側に設けられることを特徴とする請求項 8 または 9 記載の論理回路の遅延最適化方法。

【請求項 11】 フリップフロップと、組み合わせ回路とを含む論理回路の

遅延最適化方法をコンピュータに実行させるためのプログラムであって、

ラッチに変換しない前記フリップフロップを選定するラッチ化フリップフロップ選定ステップと、

前記ラッチ化フリップフロップ選定ステップにて選定されなかった前記フリップフロップを、入力側の遅延余裕を出力側に貸し出すことが可能なスルー状態を有するラッチに変換するラッチ変換ステップとを含むことを特徴とするプログラム。

【請求項 1 2】

前記フリップフロップはクロックにおける第 1 レベルから第 2 レベルへの変化時のエッジに応じて入力されるデータを保持するものであり、前記ラッチ変換ステップにおけるラッチは前記クロックが前記第 1 のレベルを示している際の入力データを出力側にスルーするものであることを特徴とする請求項 1 1 記載のプログラム。

【請求項 1 3】

前記ラッチ変換ステップにおけるラッチとは異なる時間にスルー状態となる第 2 のラッチを論理回路の所定部分に挿入するラッチ挿入ステップを含むことを特徴とする請求項 1 1 または 1 2 記載のプログラム。

【請求項 1 4】

前記ラッチ挿入ステップにおける第 2 のラッチは前記クロックが前記第 2 のレベルを示している際の入力データを出力側にスルーするものであることを特徴とする請求項 1 3 記載のプログラム。

【請求項 1 5】

前記ラッチ挿入ステップにおける第 2 のラッチは、前記ラッチ変換ステップにおけるラッチの出力側に設けられることを特徴とする請求項 1 3 または 1 4 記載のプログラム

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、論理回路の遅延最適化システムおよびその最適化方法ならびに制御

プログラムに関し、特に順序回路の遅延最適化システムおよびその最適化方法ならびに制御プログラムに関する。

【 0 0 0 2 】

【従来の技術】

論理回路の遅延最適化においては、フリップフロップやラッチに関する最適化を行う順序回路の遅延最適化手法と、フリップフロップやラッチの間に存在する組み合わせ回路部分の遅延最適化手法という 2 つの手法が存在する。本発明は順序回路の遅延最適化の方法に関する。

【 0 0 0 3 】

順序回路の遅延最適化としては、その一例が、特開平 6 - 2 9 0 2 3 2 号公報（以下、文献 1 という）に記載されているリタイミングという手法が存在する。

【 0 0 0 4 】

リタイミングに関しては、フリップフロップの位置を変更する事により、最悪遅延を改善しようとするものである。

【 0 0 0 5 】

【発明が解決しようとする課題】

しかし、この文献 1 記載の技術には、次のような欠点があった。第 1 の問題点は、文献 1 記載の技術ではフリップフロップの位置を変更してしまうので、当初の論理回路と最適化後の論理回路では、論理解析の中心となるフリップフロップの動作が変更されてしまう。したがって、当初の論理回路を設計した人が論理解析を行おうとした場合、対応が取り難いという欠点があった。

【 0 0 0 6 】

第 2 の問題点は、文献 1 記載の技術では、フリップフロップの位置を変更するだけで、遅延計算上は特に過不足が発生しないということである。

【 0 0 0 7 】

また、本発明と同様な方式として、特開 2 0 0 0 - 3 0 5 9 6 2 号公報（以下、文献 2 という）記載の技術が挙げられる。本発明と文献 2 記載の技術の違いは、文献 2 記載の技術では、入力側に遅延余裕のあるフリップフロップしかラッチ変換の対象にしないということである。

【 0 0 0 8 】

そこで、本発明の目的は上記の問題点を解消し、フリップフロップの出力遅延、セットアップ時間およびクロックスキューの低減が可能で、しかも利用者から指定されなかった全フリップフロップ（入力に遅延余裕のないフリップフロップを含む）をラッチに変換することが可能な論理回路の遅延最適化システムおよびその最適化方法ならびに制御プログラムを提供することにある。

【 0 0 0 9 】

【課題を解決するための手段】

本発明による論理回路の遅延最適化システムは、フリップフロップと、組み合わせ回路とを含む論理回路の遅延最適化システムであって、そのシステムはラッチに変換しない前記フリップフロップを選定するラッチ化フリップフロップ選定手段と、前記ラッチ化フリップフロップ選定手段にて選定されなかった前記フリップフロップを、入力側の遅延余裕を出力側に貸し出すことが可能なスルー状態を有するラッチに変換するラッチ変換手段とを含むことを特徴とする。

【 0 0 1 0 】

また本発明による論理回路の遅延最適化方法は、フリップフロップと、組み合わせ回路とを含む論理回路の遅延最適化方法であって、その方法はラッチに変換しない前記フリップフロップを選定するラッチ化フリップフロップ選定ステップと、前記ラッチ化フリップフロップ選定ステップにて選定されなかった前記フリップフロップを、入力側の遅延余裕を出力側に貸し出すことが可能なスルー状態を有するラッチに変換するラッチ変換ステップとを含むことを特徴とする。

【 0 0 1 1 】

また本発明によるプログラムは、フリップフロップと、組み合わせ回路とを含む論理回路の遅延最適化方法をコンピュータに実行させるためのプログラムであって、そのプログラムはラッチに変換しない前記フリップフロップを選定するラッチ化フリップフロップ選定ステップと、前記ラッチ化フリップフロップ選定ステップにて選定されなかった前記フリップフロップを、入力側の遅延余裕を出力側に貸し出すことが可能なスルー状態を有するラッチに変換するラッチ変換ステップとを含むことを特徴とする。

【 0 0 1 2 】

すなわち、本発明によれば、フリップフロップの出力遅延、セットアップ時間およびクロックスキューの低減が可能で、しかも利用者から指定されなかった全フリップフロップ（入力に遅延余裕のないフリップフロップを含む）をラッチに変換することが可能となる。

【 0 0 1 3 】

本発明は、論理回路内を検索し、フリップフロップを選択し、それをラッチに置き換えて、クロックスキューの遅延時間をキャンセルし、かつ入力側の遅延に関する余裕を出力側に貸し出す事で、最悪遅延の向上を達成する方法を提供するものである。

【 0 0 1 4 】

図 1 において、入力論理回路 3 1 には、本発明で遅延時間を最適化しようとする論理回路が記憶されている。回路入力部 2 2 は、入力論理回路 3 1 を読み込み、内部 DB 3 2 に設定する。遅延解析部 2 3 は、内部 DB 3 2 中の各回路素子に関する遅延計算を行う。ラッチ化 FF 選定部 2 4 は、内部 DB 3 2 中のフリップフロップにおいて、入力装置 1 でラッチ変換をしないように非ラッチ化指定されているものを探す。ラッチ変換部 2 5 は、非ラッチ化指定以外のフリップフロップをスルー状態を持つラッチに変換する。

【 0 0 1 5 】

ラッチはスルー状態を持つため、フリップフロップをラッチに変換すると最悪遅延は向上するものの、MIN 遅延補償（最小遅延補償）を満足出来なくなる。

【 0 0 1 6 】

これを満足するために、ラッチ挿入部 2 6 は、先にラッチ変換部で使用したラッチとは反対の時間でスルーを行うラッチを、回路中の適当な部分に挿入する。回路出力部 2 7 は遅延最適化後の論理回路を出力論理回路 3 3 として出力する。記録媒体 2 8 には論理回路の遅延最適化方法をコンピュータに実行させるためのプログラムが格納されている。

【 0 0 1 7 】

このようにラッチに変換した事により、クロックスキューのキャンセルと、元

のフリップフロップの入力側にあった遅延の余裕分を後段に貸し出して最悪遅延の向上を行い、また反対の時間でスルーを行うラッチを必要な部分に挿入する事でMIN遅延も満足した回路に自動変換することが出来る。

【 0 0 1 8 】

【発明の実施の形態】

以下、本発明の実施の形態について添付図面を参照しながら説明する。図1は本発明に係る論理回路の遅延最適化システムの一実施例の構成図、図2は最適化対象の論理回路の一例の構成図、図3は、図4はラッチ変換後の論理回路の一例の構成図、図5はハイアクティブ・ラッチ（HL）挿入後の論理回路の一例の構成図である。

【 0 0 1 9 】

まず、図2から図5について簡単に説明しておく。図2は最適化対象の論理回路の一例を示しており、フリップフロップ（FF）5a～5fと、組み合わせ回路6a，6bとを含む順序回路で構成されている。図3はこの回路に対し非ラッチ化指定を行った後の回路の状態を示しており、一例としてフリップフロップ（FF）5gが非ラッチ化指定されたことを示している。図4はラッチ変換後の回路の状態を示しており、一例としてフリップフロップ（FF）5b～5fがラッチ（LL）7a～7eにそれぞれ変換されたことを示している。図5はさらに一例としてハイアクティブ・ラッチ（HL）が挿入されたことを示している。

【 0 0 2 0 】

図1を参照すると、論理回路の遅延最適化システムはキーボード等の入力装置1と、プログラム制御により動作するデータ処理装置2と、情報を記憶する記憶装置3と、ディスプレイ装置等の出力装置4とを含んで構成されている。

【 0 0 2 1 】

記憶装置3は、入力論理回路31と、内部DB（データベース）32と、出力論理回路33と、ライブラリ34と、遅延制約35とを含んでいる。

【 0 0 2 2 】

入力論理回路31は、図2に示すように、フリップフロップ（FF）5と組み合わせ回路6とから構成されているが、その他に論理回路の外部とのインターフ

エイスも存在する。しかし、本実施例ではそのインターフェイスは参照されない
ので、記述を省略する。なお、組み合わせ回路 6 a, 6 b は、AND, OR, イ
ンバータ、バッファと、それらを複合した論理回路素子との集合として構成され
ている。

【 0 0 2 3 】

内部 DB 3 2 は、論理回路の情報を、データ処理装置が扱いやすいように加工
したものであり、論理回路中の各論理回路素子情報とそれらの接続情報、そして
後述する遅延計算値を格納出来るようになっている。

【 0 0 2 4 】

出力論理回路 3 3 は、最終的に変換された論理回路を保存したものである。ラ
イブラリ 3 4 は、後述する図 7 以降で表記している各論理回路素子 9、フリップ
フロップ (FF) 5、ラッチ (LL, HL) 7, 8 に関する遅延情報を格納して
いる。

【 0 0 2 5 】

遅延制約 3 5 には、設計用の遅延制約が格納されている。具体的には、設計対
象のクロックの情報 (図 8) と、論理回路の外部との入力インターフェイスにあ
たる部分の遅延値を到着時間として表示したものである。

【 0 0 2 6 】

データ処理装置 2 は、制御部 2 1 と、回路入力部 2 2 と、遅延解析部 2 3 と、
ラッチ化 FF 選定部 2 4 と、ラッチ変換部 2 5 と、ラッチ挿入部 2 6 と、回路出
力部 2 7 と、記録媒体 2 8 とを含んでいる。

【 0 0 2 7 】

制御部 2 1 は、記録媒体 2 8 に格納された遅延最適化処理のプログラムにした
がって回路入力部 2 2 と、遅延解析部 2 3 と、ラッチ化 FF 選定部 2 4 と、ラッ
チ変換部 2 5 と、ラッチ挿入部 2 6 と、回路出力部 2 7 とを呼び出す順番の制御
を行う。

【 0 0 2 8 】

回路入力部 2 2 は、入力論理回路 3 1 を読み、内部 DB 3 2 に設定する。遅延
解析部 2 3 は、内部 DB 3 2 内の各論理回路素子についての遅延計算を行う。ラ

タッチ化 F F 選定部 2 4 は、論理回路中の全てのフリップフロップ (F F) 5 に対して、入力装置 1 から指定されたラッチ化しないフリップフロップ (F F) を探索し、その情報を非ラッチ化指定として内部 D B 3 2 に設定する。ここで、図 3 のフリップフロップ (F F) 5 g が非ラッチ化指定されたとする。

【 0 0 2 9 】

ラッチ変換部 2 5 は、ラッチ化 F F 選定部 2 4 で非ラッチ化指定されなかったフリップフロップ (F F) 5 b ~ 5 f を、入力側の遅延余裕を出力側に貸し出す事が出来るようなスルー状態を持つラッチ 7 に変換する。図 4 のローアクティブ・ラッチ (L L) 7 a ~ 7 e がこれに当たる。

【 0 0 3 0 】

ラッチ挿入部 2 6 は、ラッチ変換部 2 5 で変換されたラッチ (L L) 7 の出力側の適当な位置に、先にラッチ変換部 2 5 で使用したラッチ (L L) 7 とは反対の時間でスルーを行うラッチ (ハイアクティブ・ラッチ: H L) 8 を、図 5 のように挿入する。

【 0 0 3 1 】

回路出力部 2 7 は、最終的に変換された内部 D B 3 2 を、出力論理回路 3 3 として記憶装置中に出力する。

【 0 0 3 2 】

ここで、フリップフロップ (F F) およびラッチ (L L, H L) について簡単に説明しておく。フリップフロップ (F F) は、クロックがローレベルからハイレベルへの変化時のエッジに応じて入力されるデータを保持するもの、あるいはハイレベルからローレベルへの変化時のエッジに応じて入力されるデータを保持するものである。

【 0 0 3 3 】

一方、ラッチにはローアクティブ・ラッチ (L L) とハイアクティブ・ラッチ (H L) とがある。ローアクティブ・ラッチ (L L) は、クロックがローレベルを示すときに入力されるデータを出力側にスルーする。そして、クロックがハイレベルに変化した際にはデータをロックし、その後ハイレベル中に入力データ

の変動があっても出力データは変動しない。ハイクティブ・ラッチ（HL）は、クロックがハイレベルを示すときに入力されるデータを出力側にスルーする。そして、クロックがローレベルに変化した際にはデータをロックし、その後ローレベル中に入力データの変動があっても出力データは変動しない。

【 0 0 3 4 】

図 6 は論理回路の遅延最適化処理の一例を示すフローチャート、図 7 は論理回路の具体例の一例を示す回路図、図 8 はクロックサイクルと各時刻との対応を示す説明図、図 9 は到着時間の計算例を示す図、図 1 0 はラッチ変換後の論理回路の具体例の一例を示す回路図、図 1 1 はハイクティブ・ラッチ（HL）挿入後の論理回路の具体例の一例を示す回路図である。

【 0 0 3 5 】

図 7 を参照すると、論理回路はフリップフロップ（FF）5 m ～ 5 w, 5 y と、AND ゲート 9 a, 9 c, 9 e, 9 h, 9 j, 9 m, 9 n と、OR ゲート 9 b, 9 d, 9 f, 9 g, 9 i, 9 k とを含んでいる。

【 0 0 3 6 】

図 9 を参照すると、図 7 の各構成部分に各々の到着時間が表示されている。図 1 0 を参照すると、図 7 のフリップフロップ（FF）5 o ～ 5 q およびフリップフロップ（FF）7 k がローアクティブ・ラッチ（LL）7 f ～ 7 i およびローアクティブ・ラッチ（LL）7 k に変換されている。図 1 1 を参照すると、図 1 0 の論理回路にハイクティブ・ラッチ（HL）8 e ～ 8 i が挿入されている。

【 0 0 3 7 】

次に、図 1 及び図 6 を参照して本実施例の動作について詳細に説明する。入力装置 1 により指定された入力論理回路 3 1 は、回路入力部 2 2 により、内部 DB 3 2 に変換される（図 6 のステップ A 1）。入力論理回路 3 1 として、図 7 の論理回路を例に取り、以下の説明を行う。また、遅延制約 3 5 には、設計用の遅延制約が格納されている。具体的には、設計対象のクロックの情報（図 8）と、論理回路の入力部に対し、外部のフリップフロップからその部分までに要する遅延値を到着時間として記述している。

【 0 0 3 8 】

遅延解析部 2 3 は、内部 DB 3 2 に対して遅延計算用の遅延制約の設定を行う（ステップ A 2）。具体的には、遅延制約は、到着時間と呼ばれ、論理回路の外部とのインターフェイスにあたる入力には遅延制約 3 5 より対応する値を探し設定し、フリップフロップ（FF）5 の出力には、フリップフロップの出力遅延値をライブラリ 3 4 から検索し設定する。また、クロックの目標周期を T_{ns} として記憶する。

【 0 0 3 9 】

本実施例では一例として、フリップフロップ（FF）の出力遅延値は 1 ns 、セットアップ時間も 1 ns 、クロックサイクルは 8 ns 、クロックスキューは 1 ns 、ラッチのスルー遅延は 1 ns として説明を行う。従って、フリップフロップ（FF）の出力には到着時間として 1 ns が設定される。

【 0 0 4 0 】

次に遅延解析部 2 3 は、全ての論理回路素子 9 の到着時間の計算を行う（ステップ A 3）。到着時間に関しては、論理回路の外部入力とフリップフロップ（FF）5 の出力から、出力方向に向かって、各論理回路素子 9 に対して、式 1 を用いて計算される。

【 0 0 4 1 】

論理回路素子の到着時間 $= \max$ （各入力側の論理回路素子の到着時間 + 入力 PIN から出力 PIN の遅延時間） （1）

ここで、組み合わせ回路素子の入力 PIN から出力 PIN の遅延時間は、ライブラリ 3 4 を用いて、各論理回路素子毎に計算されるが、本実施例の説明では、一律 2 ns とする。

【 0 0 4 2 】

図 9 に、この計算の結果を示す。各論理回路素子 9 内に記入されている数値は、到着時間を示している。フリップフロップ（FF）5 に関しては、上側にフリップフロップ（FF）5 の出力ピンの到着時間を、下側に入力からの到着時間を示す。例えば、フリップフロップ（FF）5 o に関しては、出力側の到着時間は、前記ライブラリに格納された値 1 ns が設定されている。

【 0 0 4 3 】

論理回路素子 9 f の到着時間は、9 f の遅延時間が 2 n s なので、式 (1) より、

$$\text{到着時間} = \max (1 + 2, 1 + 2) = 3 \text{ n s} \quad (2)$$

と計算される。このような処理を出力に向かって計算する事で全ての論理回路素子 9 とフリップフロップ (F F) 5 の到着時間が求められる。

【 0 0 4 4 】

ラッチ化 F F 選定部 2 4 は、入力装置から設定される非ラッチ化の指定を設定する。このフリップフロップ (F F) はラッチ化しない (ステップ A 4)。ここで、フリップフロップ (F F) 5 m, 5 n, 5 s, 5 v, 5 w, 5 y が非ラッチ化指定されたと仮定する。

【 0 0 4 5 】

ラッチ変換部 2 5 は、非ラッチ化指定されていないフリップフロップ (5 o, 5 p, 5 q, 5 r, 5 t, 5 u) を、入力側の遅延余裕を出力側に貸し出す事が出来るようなスルー状態を持つラッチ (図 1 0 のフリップフロップ 7 f, 7 g, 7 h, 7 i, 7 j, 7 k) に変換する (ステップ A 5)。

【 0 0 4 6 】

ここで、入力側の遅延余裕を、出力側に貸し出す事とは、具体的には次のようなことを表す。フリップフロップ (F F) 5 m, 5 n から出る信号が、時刻 A (図 8) にフリップフロップ (F F) 5 m, 5 n を出発するとする。論理回路素子 9 a の到着時間が 3 n s なので、フリップフロップ (F F) 5 o には、時刻 B より先に到着している事になる。しかし、フリップフロップ (F F) 5 o は、次のクロックの立ち上がりである時刻 C から、入力側の信号の変化を出力側へ伝播させる。

【 0 0 4 7 】

ここで、フリップフロップ (F F) 5 o を、クロックがローレベルの時にスルーし、クロックの立ち上がりでホールドするようなローアクティブ・ラッチ (L L) 7 f に変換すると、時刻 A でフリップフロップ (F F) 5 m, 5 n を出発した信号は、時刻 B になるとラッチ (L L) 7 f の出力側に信号伝播を開始する。

【 0 0 4 8 】

つまり、時刻BとCの時間差だけ早く伝播する事が可能となる。

【0049】

ラッチ挿入部26は、先にラッチ変換部25で使用したラッチ（LL）7とは反対の時間でスルーを行うラッチ（HL）8を、回路中のクロックの目標周期T近くの個所に挿入する（ステップA6）（図11参照）。

【0050】

図11を参照すると、ANDゲート9eとフリップフロップ（FF）5sとの間にハイアクティブ・ラッチ8eが、ANDゲート9hとローアクティブ・ラッチ（LL）7jとの間にハイアクティブ・ラッチ8gが、ORゲート9iとANDゲート9jとの間にハイアクティブ・ラッチ8fが、ORゲート9iとANDゲート9jとの間にハイアクティブ・ラッチ8fが、ローアクティブ・ラッチ（LL）7jとANDゲート9jとの間にハイアクティブ・ラッチ8hが、インバータ9lとANDゲート9mとの間にハイアクティブ・ラッチ8iがそれぞれ挿入されている。

【0051】

このハイアクティブ・ラッチ（HL）8を挿入する理由は次の通りである。図10を参照すると、ローアクティブ・ラッチ（LL）7fをスルーした信号は、ANDゲート9eを経由し、フリップフロップ（FF）5sに到着する。ここで、本来、図8の時刻Cでフリップフロップ（FF）5oを出す信号を、時刻Eでフリップフロップ（FF）5sは受け取るはずであったが、ローアクティブ・ラッチ（LL）7fがスルーしているため、時刻Cで受け取ってしまう事がある。

【0052】

これを防ぐために、反対の時間でスルーを行うハイアクティブ・ラッチ（HL）8を適当な位置に挿入する必要がある（MIN遅延補償）。そこで、ハイアクティブ・ラッチ（HL）8eをANDゲート9eとフリップフロップ（FF）5sとの間に挿入することにより、その信号は時刻Cから時刻Dまではハイアクティブ・ラッチ（HL）8eをスルーすることになるため、フリップフロップ（FF）5sはその信号を時刻Eに受け取ることが可能となる。

【0053】

そして、回路出力部 2 7 が、出力論理回路 3 3 として出力する（ステップ A 7）。

【 0 0 5 4 】

例えば、図 9（従来の論理回路）の場合のフリップフロップの利用時は、
 クロック周期＝フリップフロップ（FF）への到着時間の最大値＋フリップフロップ（FF）のセットアップ時間＋クロックスキュー＝ 9 + 1 + 1 = 1 1 n s

となる。なお、フリップフロップ（FF）への到着時間の最大値は、図 9 を参照するとフリップフロップ（FF）5 u の 9 n s であることが分かる。

【 0 0 5 5 】

それに比べて、図 1 1 の本発明の論理回路では、図 1 2 のようにクロック周期が 8 n s で動作する。図 1 2 は本発明の信号伝送のタイミングチャートである。

【 0 0 5 6 】

すなわち、図 1 1 および図 1 2 を参照すると、クロックの立ち上がり時刻 A に本論理回路に入力された信号はフリップフロップ（FF）5 m（遅延値 1 n s）、AND ゲート 9 a（遅延値 2 n s）を通過し、クロックの立ち下がり時刻 B にローアクティブ・ラッチ（LL）7 f（遅延値 1 n s）、OR ゲート 9 f（遅延値 2 n s）、OR ゲート 9 i（遅延値 2 n s）、ハイアクティブ・ラッチ（HL）8 f（遅延値 1 n s）、AND ゲート 9 j（遅延値 2 n s）を通過し、クロックの立ち下がり時刻 D に OR ゲート 9 k（遅延値 2 n s）、ローアクティブ・ラッチ（LL）7 k（遅延値 1 n s）、インバータ 9 l（遅延値 2 n s）、ハイアクティブ・ラッチ（HL）8 i（遅延値 1 n s）、AND ゲート 9 m（遅延値 2 n s）を通過し、クロックの立ち下がり時刻 F に AND ゲート 9 n（遅延値 2 n s）を通過する。そして、セットアップ時間（1 n s）とクロックスキュー（1 n s）の時間を加算すると正確にクロックの立ち上がり時刻 G となる。

【 0 0 5 7 】

また、本発明と同様な方式として、前述の文献 2 記載の技術があるが、これは入力に余裕のあるフリップフロップしかラッチに変換しないので、図 7 の回路は図 1 3 のようになる。図 1 3 は従来のラッチ変換を採用した論理回路の一例の回

路図である。

【 0 0 5 8 】

図 1 3 を参照すると、入力側に遅延余裕のないフリップフロップ (F F) 5 u はラッチ変換されず、その結果フリップフロップ (F F) 5 u の出力側のパスがクリティカルパスになってしまう。その時のクロック周期は、

クロック周期 = フリップフロップ (F F) の遅延 + 組み合わせ回路の遅延 + フリップフロップのセットアップ時間 + クロックスキュー = 1 + 6 + 1 + 1 = 9 n s

となり、本方式の 8 n s より大きい値になってしまう。

【 0 0 5 9 】

次に、本発明に係る論理回路の遅延最適化方法をコンピュータに実行させるためのプログラムについて説明する。図 1 を参照すると、データ処理装置 2 は記録媒体 2 8 を含んでいる。この記録媒体 2 8 には図 6 にフローチャートで示す論理回路の遅延最適化処理のプログラムが格納されている。制御部 2 1 は記録媒体 2 8 からこのプログラムを読み出し、そのプログラムにしたがって回路入力部 2 2 から回路出力部 2 7 までを制御する。その制御の内容については既に述べたので説明を省略する。

【 0 0 6 0 】

【発明の効果】

以上説明したように本発明による論理回路の遅延最適化システムによれば、ラッチに変換しない前記フリップフロップを選定するラッチ化フリップフロップ選定手段と、前記ラッチ化フリップフロップ選定手段にて選定されなかった前記フリップフロップを、入力側の遅延余裕を出力側に貸し出すことが可能なスルー状態を有するラッチに変換するラッチ変換手段とを含むため、フリップフロップの出力遅延、セットアップ時間およびクロックスキューの低減が可能で、しかも利用者から指定されなかった全フリップフロップ (入力に遅延余裕のないフリップフロップを含む) をラッチに変換することが可能となる。

【 0 0 6 1 】

また本発明による論理回路の遅延最適化方法およびプログラムも上記論理回路

の遅延最適化システムと同様の効果を奏する。

【 0 0 6 2 】

すなわち、本発明によれば一部のフリップフロップをラッチに変更することにより、最悪遅延を改善する事が出来る。その理由は、フリップフロップをラッチに変更する事で、ラッチのスルーを利用し、クロックスキューのキャンセルと、フリップフロップの入力側にあった遅延の余裕を、フリップフロップの出力側に貸し出す事が出来るためである。

【図面の簡単な説明】

【図 1】

本発明に係る論理回路の遅延最適化システムの一実施例の構成図である。

【図 2】

最適化対象の論理回路の一例の構成図である。

【図 3】

非ラッチ化指定後の論理回路の一例の構成図である。

【図 4】

ラッチ変換後の論理回路の一例の構成図である。

【図 5】

ハイアクティブ・ラッチ（HL）挿入後の論理回路の一例の構成図である。

【図 6】

論理回路の遅延最適化処理の一例を示すフローチャートである。

【図 7】

論理回路の具体例の一例を示す回路図である。

【図 8】

クロックサイクルと各時刻との対応を示す説明図である。

【図 9】

到着時間の計算例を示す図である。

【図 1 0】

ラッチ変換後の論理回路の具体例の一例を示す回路図である。

【図 1 1】

ハイアクティブ・ラッチ（H L）挿入後の論理回路の具体例の一例を示す回路図である。

【図 1 2】

本発明の信号伝送のタイミングチャートである。

【図 1 3】

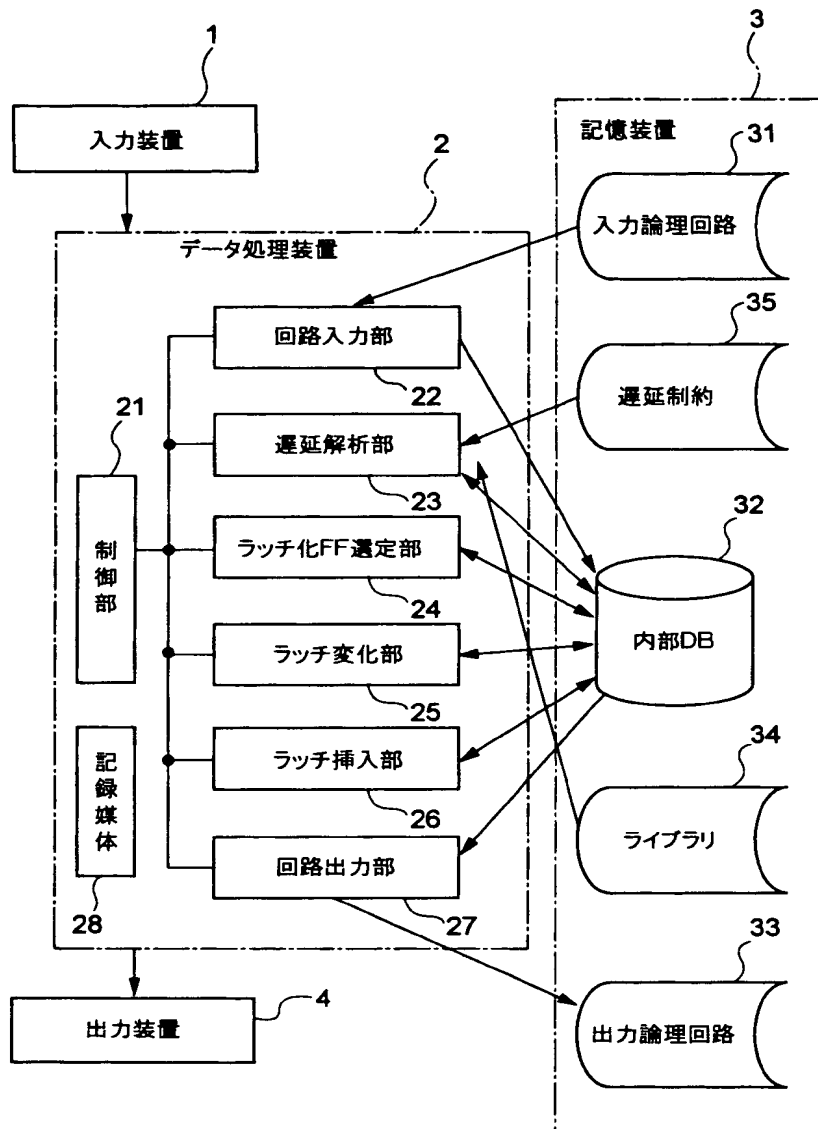
従来 of ラッチ変換を採用した論理回路の一例の回路図である。

【符号の説明】

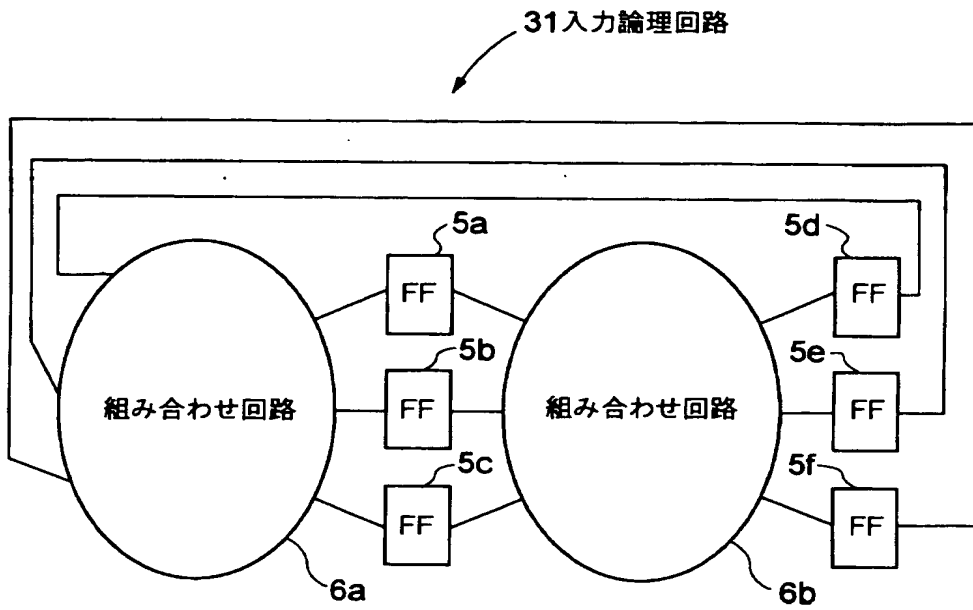
- 1 入力装置
- 2 データ処理装置
- 3 記憶装置
- 4 出力装置
- 2 1 遅延制約
- 2 2 回路入力部
- 2 3 遅延解析部
- 2 4 ラッチ化 F F 選定部
- 2 5 ラッチ変換部
- 2 6 ラッチ挿入部
- 2 7 回路出力部
- 2 8 記録媒体
- 3 1 入力論理回路
- 3 2 入力論理回路
- 3 3 出力論理回路
- 3 4 ライブラリ
- 3 5 遅延制約

【書類名】 図面

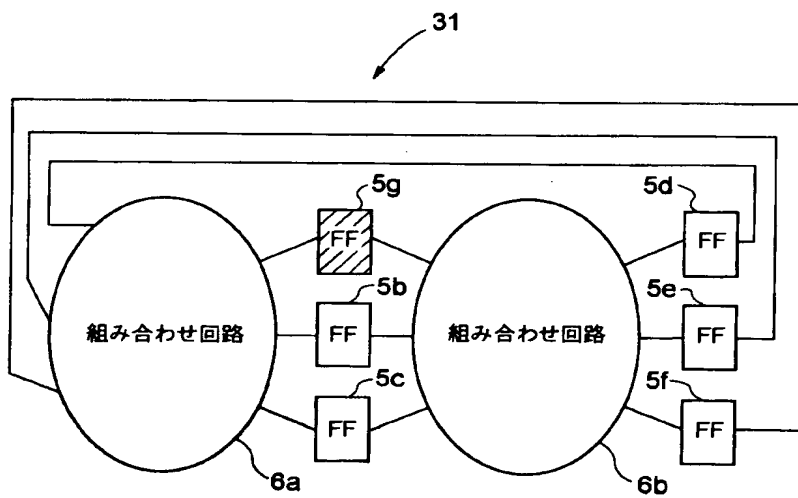
【図 1】



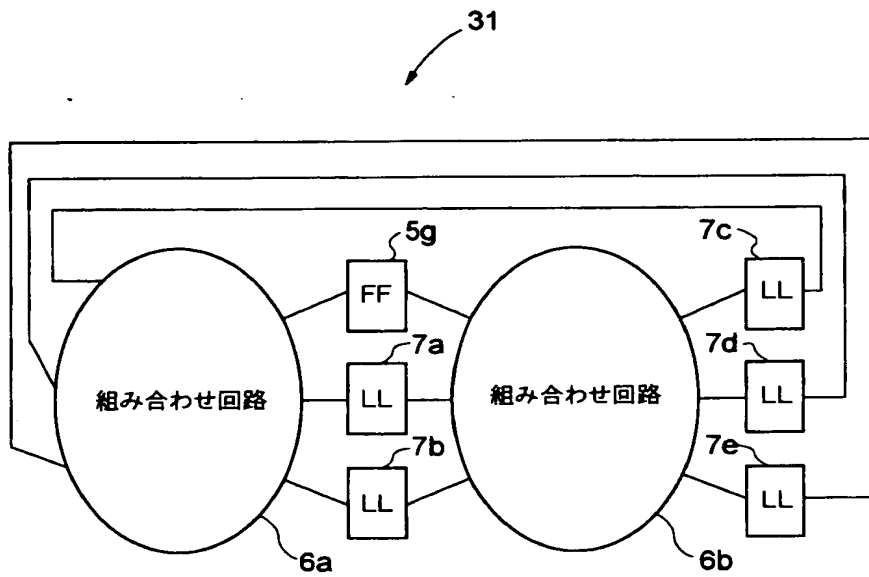
【図 2】



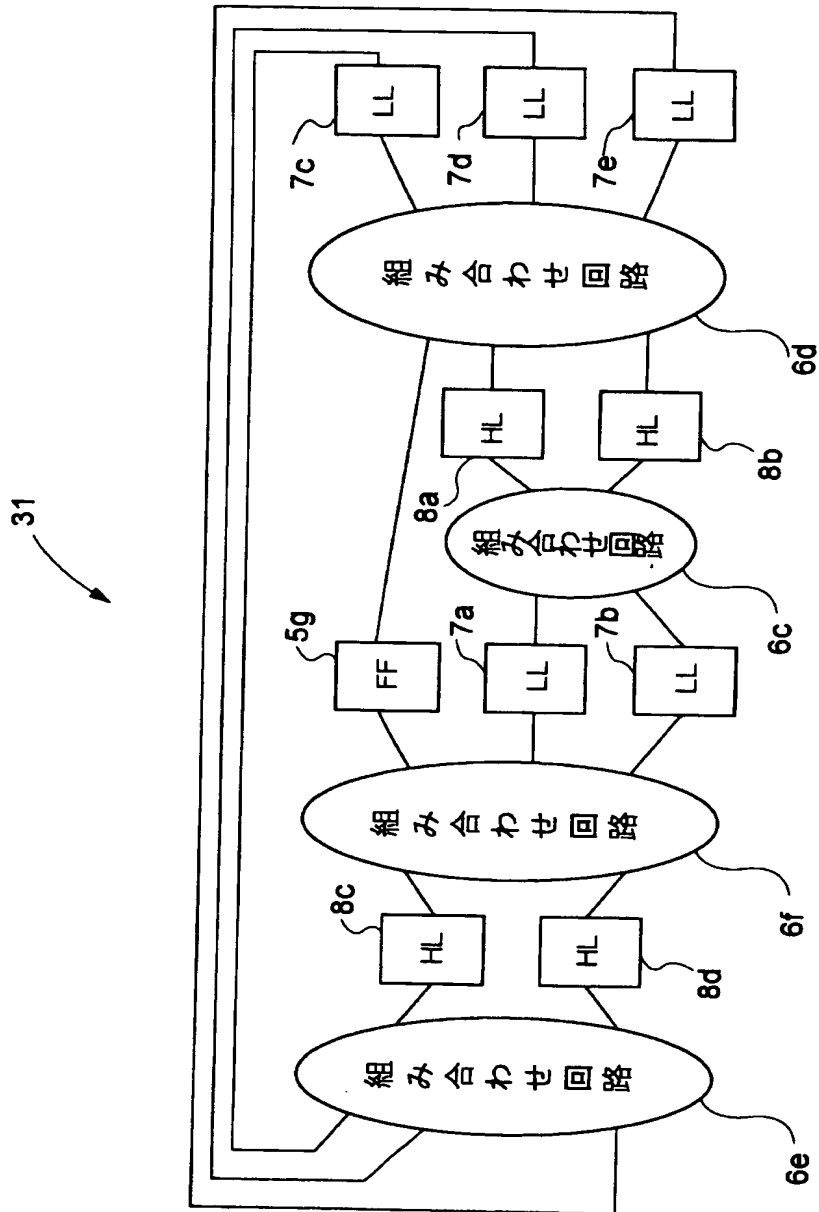
【図 3】



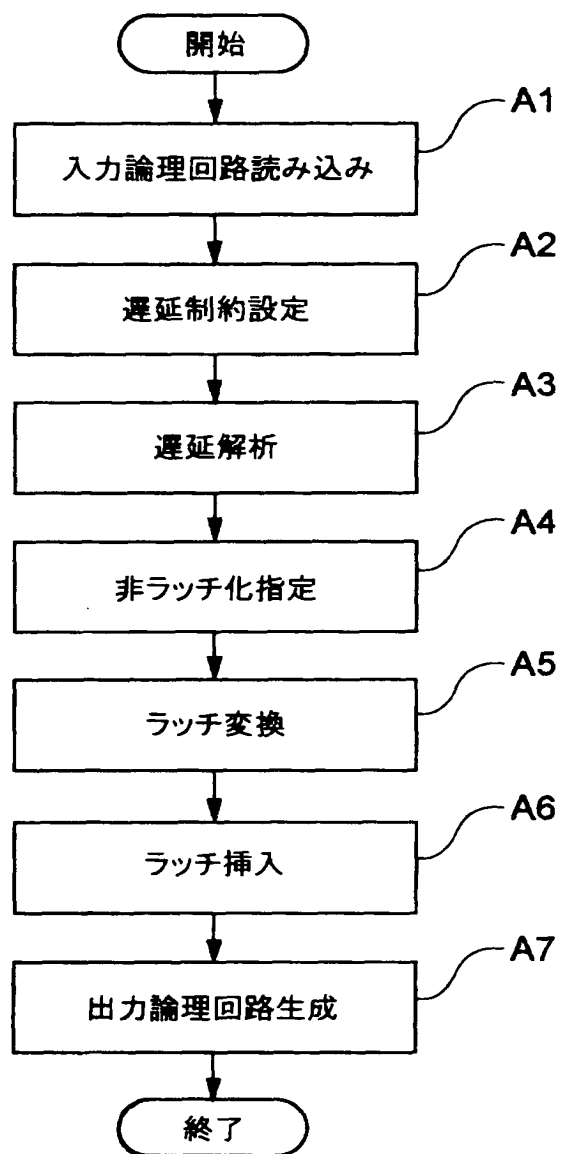
【図 4】



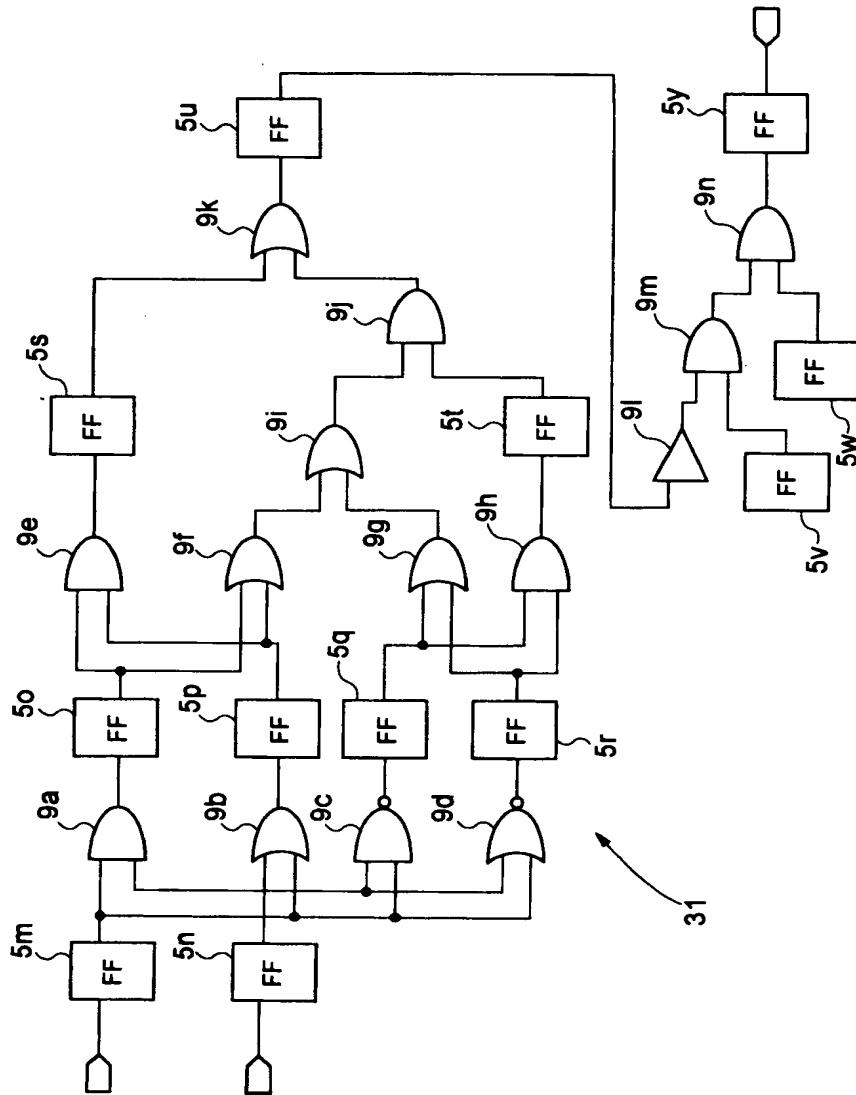
【図 5】



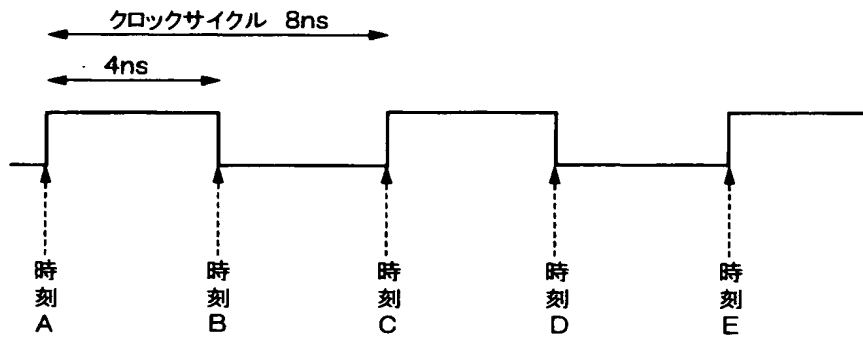
【図 6】



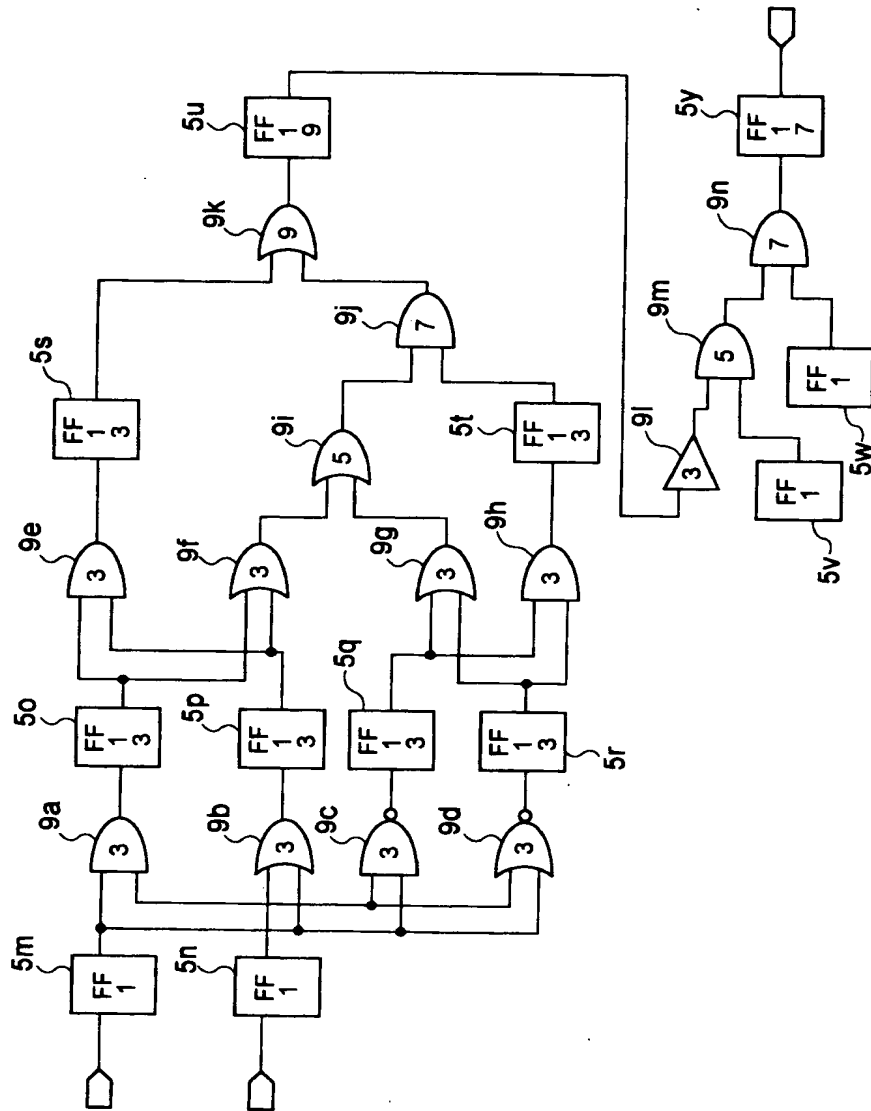
【図 7】



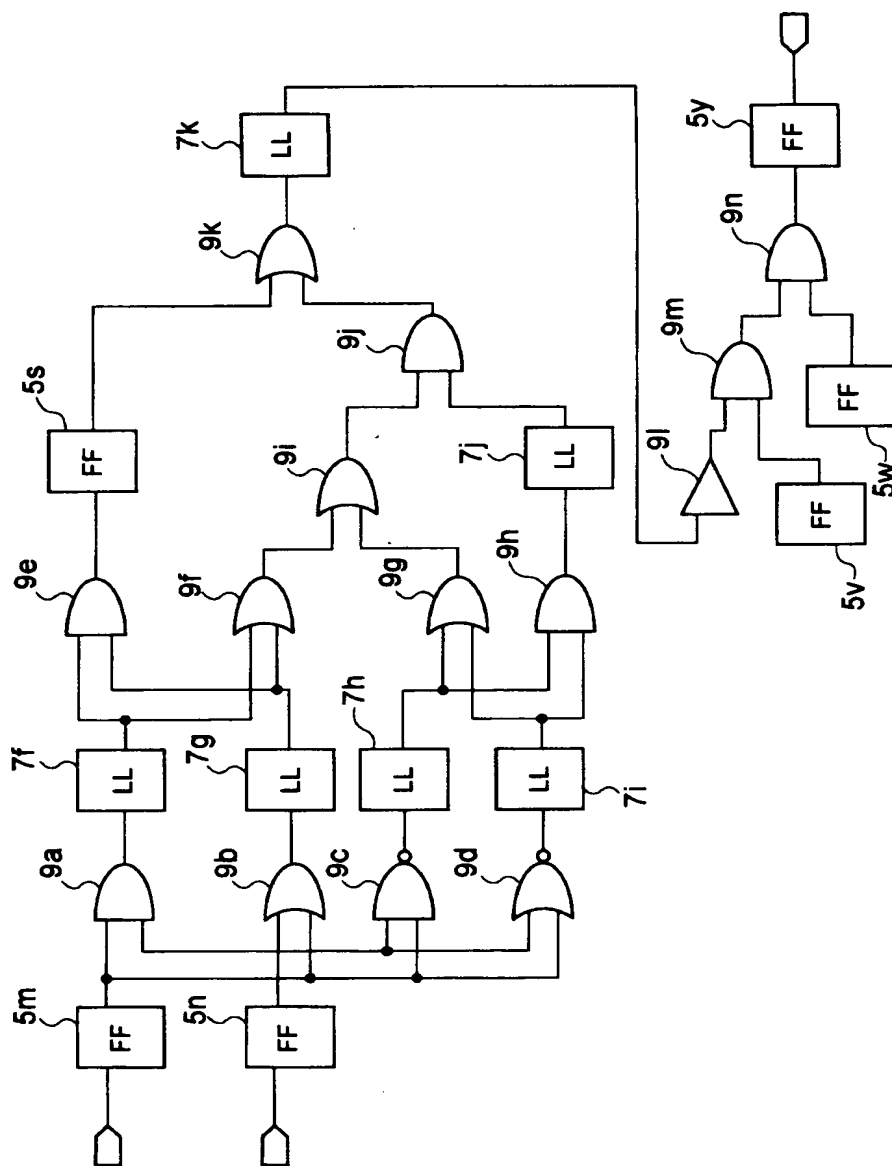
【図 8】



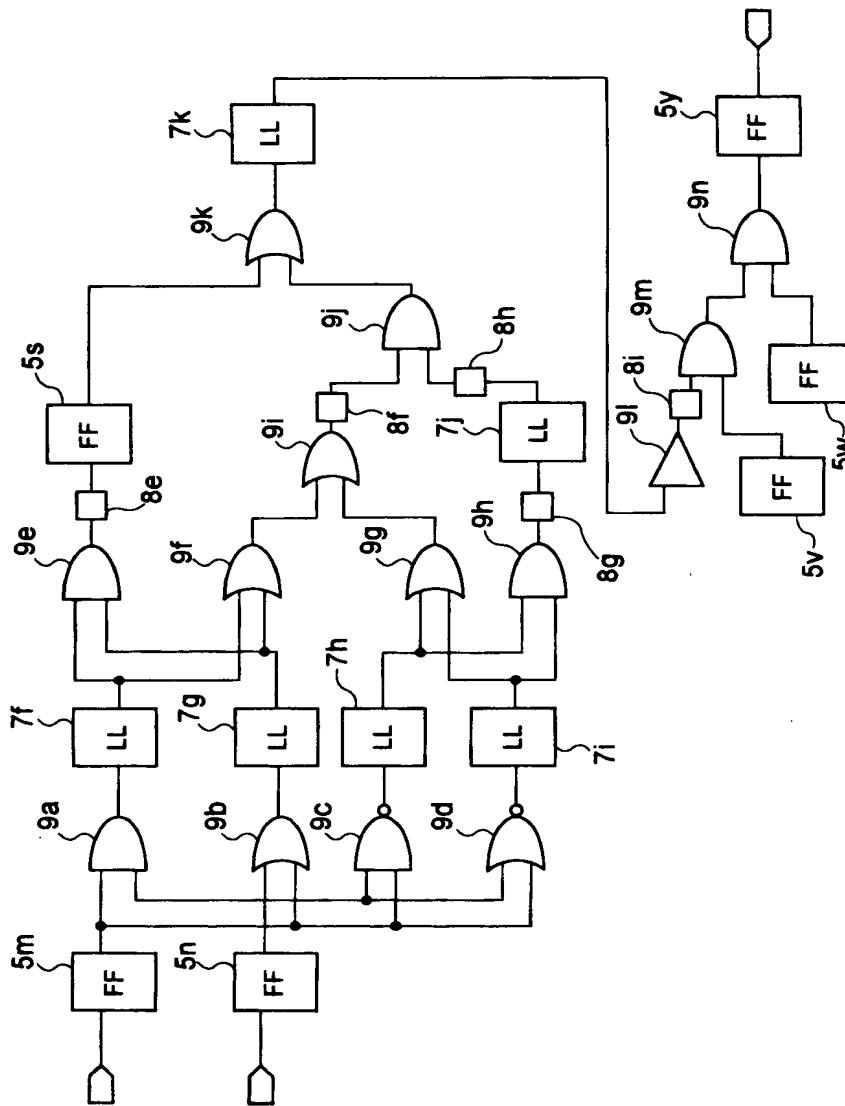
【図 9】



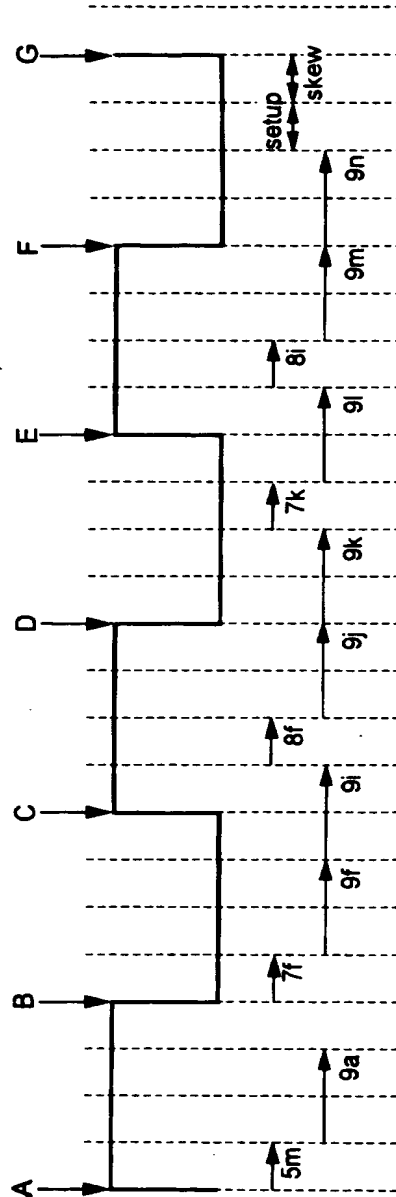
【図 10】



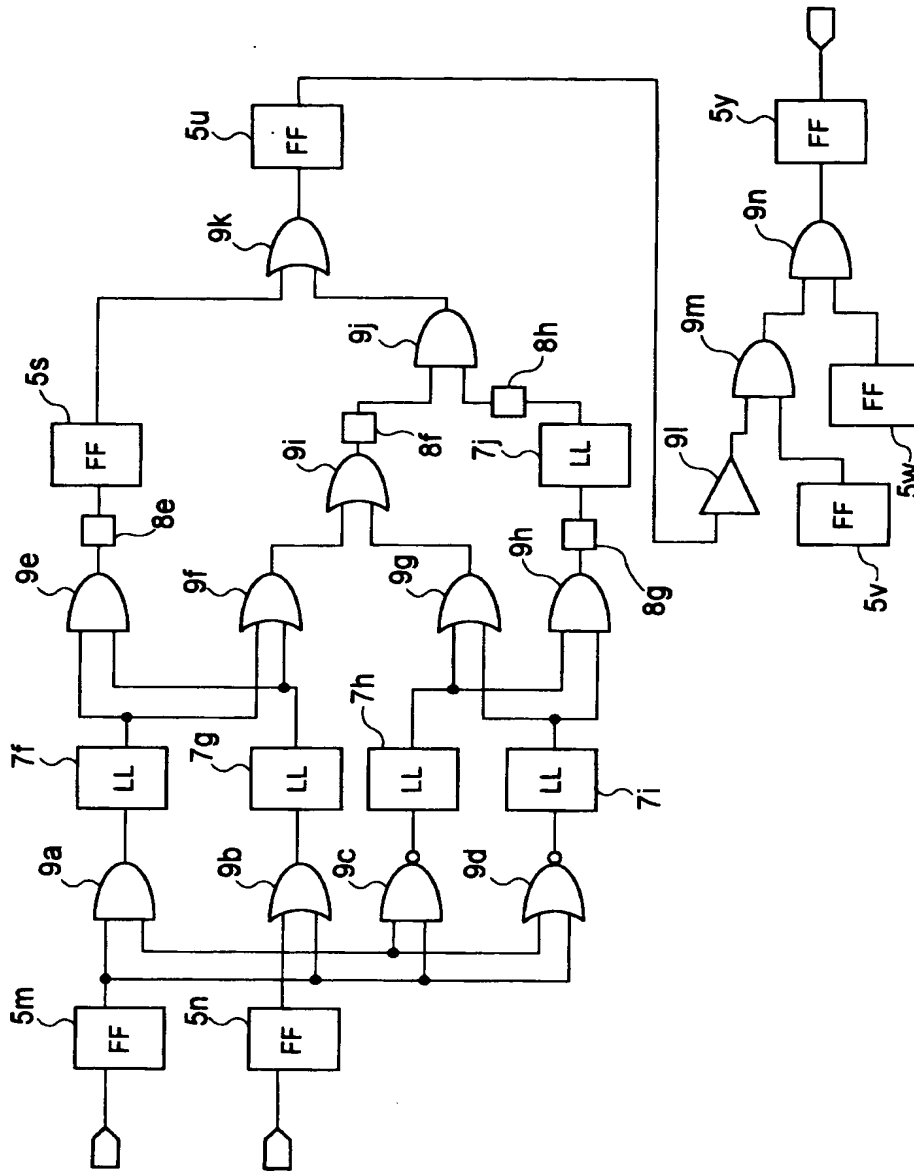
【図 11】



【図 12】



【図 13】



【書類名】 要約書

【要約】

【課題】 フリップフロップの出力遅延、セットアップ時間およびクロックスキューの低減。

【解決手段】 回路入力部 2 2 および遅延解析部 2 3 を介して入力論理回路 3 1 内を検索し、ラッチ化 F F 選定部 2 4 にてラッチ化しないフリップフロップを選択し、ラッチ化しないフリップフロップ以外のフリップフロップをラッチ変換部 2 5 にてをラッチに置き換えて、クロックスキューの遅延時間をキャンセルし、かつ入力側の遅延に関する余裕を出力側に貸し出す事で、最悪遅延の向上を達成する。

【選択図】 図 1

出 願 人 履 歴 情 報

識別番号 [0 0 0 0 0 4 2 3 7]

1. 変更年月日 1 9 9 0 年 8 月 2 9 日

[変更理由] 新規登録

住 所 東京都港区芝五丁目 7 番 1 号

氏 名 日本電気株式会社